Multithreaded Data/Context Flow Processing Architecture

ABSTRACT

Multithreaded data-flow processing is achieved by flowing data and context (thread) identification tokens through specialized cores (functional blocks, intellectual property). Each context identification token defines the identity of a context and associated context parameters affecting the processing of the data tokens. Parameter values for different contexts are stored in a distributed manner throughout the cores. Upon a context switch, only the identity of the new context is propagated. The parameter values for the new context are retrieved from the distributed storage locations. Different cores of the system and different pipestages within a core can work simultaneously in different contexts. The described architecture does not require long propagation distances for parameters upon context switches, or that an entire pipeline finish processing in one context before starting processing in another. The system is effectively controlled by the flow of data and context identification tokens therethrough. No master context controller is needed.